Een brede computationele grammatica voor het Nederlands

Leonoor van der Beek, Gosse Bouma en Gertjan van Noord*

Abstract

We present a wide-coverage computational parser and grammar for Dutch. The grammar is developed within the tradition of Head-driven Phrase Structure Grammar and provides detailed and linguistically motivated accounts for most syntactic phenomena in Dutch. A broad overview of the grammar and lexicon is given, as well as a more detailed discussion of the analysis of comparatives and partitive constructions. Parsing and disambiguation uses statistical information, which is derived in part from a syntactically annotated tree-bank. The accuracy and coverage of the grammar is evaluated on representative portions of the tree-bank and a number of problematic constructions for the current grammar are identified.

1 Inleiding

In dit artikel geven we een overzicht van de Alpino-grammatica, een brede computationele grammatica voor het Nederlands die het mogelijk maakt grote hoeveelheden tekst automatisch van een gedetailleerde syntactische analyse te voorzien.[1] De grammatica onderscheidt zich van theoretische grammatica's doordat ze nadrukkelijk is ontworpen met als doel zoveel mogelijk van de syntactische structuren te beschrijven die in vrije tekst worden aangetroffen. Ze onderscheidt zich van de meeste robuuste, *wide-coverage*, computationele grammatica's doordat de grammatica taalkundig gemotiveerd en handmatig ontwikkeld is, en doordat ook minder eenvoudige aspecten van de syntaxis (zoals de analyse van vraagzinnen of van kruisende afhankelijkheden in *verb raising*-constructies) beschreven worden. Door ontwikkeling en evaluatie van de Alpino-grammatica proberen we inzicht te krijgen in de vraag in hoeverre het mogelijk is met een taalkundig gemotiveerde, handgeschreven, grammatica, syntactische analyse uit te voeren die robuust, accuraat, en efficiënt is.

De Alpino-grammatica maakt in ruime mate gebruik van inzichten uit Head-driven Phrase Structure Grammar (Pollard & Sag 1994), in het bijzonder van de variant in Sag (1997), waarin gedetailleerde regels worden gedefinieerd in termen van algemene structuren en principes. In paragraaf 2 introduceren we de theoretische uitgangspunten en het formalisme van de grammatica. In paragraaf 3 geven we een overzicht van het lexicon en de regels die samen de Alpino-grammatica vormen.

Parallel aan het ontwikkelen van de grammatica is begonnen met de opbouw van een syntactisch geannoteerd corpus. Zo'n corpus is om een tweetal redenen van groot belang. Ten eerste wordt de Alpino-grammatica ontwikkeld door meerdere personen, gedurende een periode van tenminste enkele jaren. De omvang en complexiteit van de grammatica maakt dat handmatige evaluatie en controle

*We danken de beoordelaars van een eerdere versie van dit artikel voor hun gedetailleerde commentaar. Adres van de auteurs: Rijksuniversiteit Groningen, afdeling Alfa-informatica, Postbus 716, 9700 AS Groningen. e-mail: vdbeek@let.rug.nl, gosse@let.rug.nl, vannoord@let.rug.nl

[1] De Alpino-grammatica is ontwikkeld binnen het NWO Pionier-project *Algorithms for Linguistic Processing*.

op fouten moeizaam en arbeidsintensief wordt. Een geannoteerd corpus biedt de mogelijkheid om objectief te bepalen of veranderingen in de grammatica leiden tot een verbetering of niet. Ten tweede kent de grammatica aan iedere zin van enige omvang een groot aantal lezingen toe, een probleem dat met het groeien van de grammatica alleen maar toeneemt. Het kiezen van de juiste analyse is daarom een probleem. De Alpino-grammatica maakt gebruik van statistische modellen om de meest waarschijnlijke analyse van een zin te bepalen. Training en, met name, evaluatie van zulke modellen vereisen een geannoteerd corpus.

Binnen het project Corpus Gesproken Nederlands (CGN) (Oostdijk 2000) zijn richtlijnen ontwikkeld voor de syntactische annotatie van het Nederlands. Deze annotatierichtlijnen worden, op enkele kleine verschillen na, ook gehanteerd bij de opbouw van het Alpino-corpus. In tegenstelling tot het CGN bestaat het Alpino-corpus uit geschreven materiaal (ontleend aan het Eindhoven-corpus (Uit den Boogaart 1975)). In paragraaf 4 bespreken we hoe de Alpino-grammatica kan worden gebruikt als hulpmiddel bij het annotatieproces en hoe statistische desambiguatie verloopt.

In paragraaf 5 presenteren we een methode om de resultaten van syntactische analyse en desambiguatie te evalueren aan de hand van het geannoteerde corpus. We presenteren enkele kwantitatieve resultaten die een indruk geven van de mate waarin automatische syntactische analyse van vrije tekst succesvol is. Naar aanleiding van deze resultaten kunnen we bovendien een overzicht geven van fenomenen in het corpus die voor de huidige versie van de grammatica nog problematisch zijn.

In dit artikel bespreken we de grammatica die door het Alpino systeem wordt gebruikt. We gaan verder niet in op het gebruikte *parseeralgoritme* dat op grond van deze grammatica en een gegeven zin de analyses uitrekent, maar we volstaan met de opmerking dat de Alpino-parser een efficiënte Prolog implementatie van een *left-corner parser* is, zoals uitgebreid beschreven in Van Noord (1997). Deze parser construeert in eerste instantie een *parse forest*: een compacte representatie van alle mogelijke syntactische analyses. Op basis van het desambiguatiemodel probeert het systeem hieruit vervolgens de juiste syntactische analyse te selecteren. Het desambiguatiemodel wordt besproken in paragraaf 4.

2 Uitgangspunten

Head-driven Phrase Structure Grammar (Pollard & Sag 1994, Sag & Wasow 1999) is een taalkundige theorie die tracht om formeel precieze beschrijvingen en verklaringen te geven van taalkundige fenomenen. Omdat computationele overwegingen bij het ontwerp van het formalisme niet buiten beschouwing zijn gebleven, is het een theorie die ook binnen de computationele taalkunde aanzien geniet. Brede computationele grammatica's die gebaseerd zijn op HPSG bestaan onder andere voor het Engels, Duits, en Japans. HPSG-analyses van aspecten van de Nederlandse syntaxis zijn onder andere te vinden in Van Noord & Bouma (1994), Van Eynde (1996, 1999) en Bouma (2000).

HPSG maakt gebruik van *attribute-value matrices* (AVM's) in de definitie van de grammatica. In het woordenboek worden woorden gekoppeld aan een AVM die de lexicale eigenschappen van het betreffende woord representeren. De woorden *vrienden* en *vraag* in figuur 1 zijn van het type *noun*. Hieruit volgt dat bijvoorbeeld het attribuut NFORM is gedefinieerd. De waarde [NFORM *norm*] onderscheidt 'normale' zelfstandige naamwoorden van expletieven als *het* en *er*. Voor het type *verb* is VFORM gedefinieerd. Het attribuut SC bevat een lijst van elementen die als complement bij dit woord kunnen optreden. De infiniefvorm van het werkwoord *achtervolgen* selecteert bijvoorbeeld een direct object.[2] Het zelfstandig naamwoord *vraag* kan combineren met een bijzin (in de vorm van een indirecte vraag of ingeleid door *of*) als complement. Omdat dit complement geëxtraponeerd kan

[2]De finiete vorm van een *achtervolgen* onderscheidt zich onder andere van de niet-finiete vormen doordat SC naast een direct object ook een element bevat dat correspondeert met het subject.

a. *vrienden*:

$$\begin{bmatrix} \textit{noun} & \\ \text{NFORM} & \textit{norm} \\ \text{SC} & \langle\,\rangle \\ \text{EXTRA} & \langle\,\rangle \\ \text{DT} & \begin{bmatrix} \text{CAT} & \textit{n} \\ \text{HD} & \textit{vriend} \\ \text{MODS} & \langle\,\rangle \end{bmatrix} \end{bmatrix}$$

b. *vraag*:

$$\begin{bmatrix} \textit{noun} & \\ \text{NFORM} & \textit{norm} \\ \text{SC} & \langle\,\rangle \\ \text{EXTRA} & \left\langle \begin{bmatrix} \textit{sbar} & \\ \text{C\_TYPE} & \textit{ind-q} \vee \textit{of} \\ \text{DT} & \boxed{1} \end{bmatrix} \right\rangle \\ \text{DT} & \begin{bmatrix} \text{CAT} & \textit{n} \\ \text{HD} & \textit{vraag} \\ \text{VC} & \boxed{1} \\ \text{MODS} & \langle\,\rangle \end{bmatrix} \end{bmatrix}$$

c. *achtervolgen*:

$$\begin{bmatrix} \textit{verb} & \\ \text{VFORM} & \textit{inf} \\ \text{SC} & \left\langle \begin{bmatrix} \textit{noun} & \\ \text{CASE} & \textit{acc} \\ \text{DT} & \boxed{2} \end{bmatrix} \right\rangle \\ \text{EXTRA} & \langle\,\rangle \\ \text{DT} & \begin{bmatrix} \text{CAT} & \textit{v} \\ \text{HD} & \textit{achtervolg} \\ \text{OBJ1} & \boxed{2} \\ \text{MODS} & \langle\,\rangle \end{bmatrix} \end{bmatrix}$$

Figuur 1: Voorbeelden van woorden en de bijbehorende attribute-value matrices.

worden staat het op EXTRA in plaats van op SC. Het attribuut DT representeert een *dependency tree*, een weergave van de grammaticale relaties binnen de constituent waarvan dit woord het hoofd is. De rol van dit attribuut wordt aan het einde van deze paragraaf besproken.

HPSG wordt vaak gepresenteerd als een radicaal lexicalistische theorie, dat wil zeggen, als een theorie die gebruik maakt van een klein aantal algemene regelschema's in combinatie met een rijk gestructureerd lexicon. In Sag (1997) wordt een variant van HPSG voorgesteld waarin ruimte is voor constructiespecifieke regels. Zulke regels zijn vooral nuttig om aspecten van de grammatica te beschrijven die niet gemakkelijk in termen van algemene regels en specifieke lexicale elementen te beschrijven zijn, zoals bijvoorbeeld de syntaxis van relatieve zinnen. Het gevaar dat hierdoor generalisaties worden gemist wordt bezworen door regels te definiëren als instanties van algemene structuren en principes. In de Alpino-grammatica is voor dezelfde aanpak gekozen. Het feit dat parseren op basis van specifieke regels vooralsnog efficiënter is dan parseren met algemene regelschema's is een belangrijk bijkomend voordeel.

Vrijwel alle regels in de Alpino-grammatica zijn instanties van een zogenaamde *headed structure*.

Een *headed structure* bestaat uit een moederknoop, een dochter die fungeert als hoofd, en nul of meer andere dochters. Iedere *headed structure* voldoet aan de volgende principes:

- **Head-feature principle**: De attributen die als HEAD features zijn gedefinieerd in de grammatica, worden op de moeder en de *head*-dochter geünificeerd.[3]

- **Valence principle**: De AVM van een eventuele complementdochter dient te unificeren met het eerste element op SC van het hoofd. De SC-waarde van de moeder is de SC-waarde van het hoofd, eventueel minus het element dat correspondeert met een complement dochter.

- **Filler Principle**: De AVM van een eventuele *filler*-dochter dient te unificeren met het eerste element op SLASH van het hoofd. De SLASH-waarde van de moeder is de de SLASH-waarde van het hoofd, eventueel minus het element dat correspondeert met een *filler*-dochter.[4]

- **Extraposition Principle**: De AVM van een eventuele geëxtraponeerde dochter dient te unificeren met het eerste element op EXTRA van het hoofd. De EXTRA-waarde van de moeder is de *concatenatie* van de EXTRA-waardes van alle dochters, eventueel minus het element dat correspondeert met een geëxtraponeerde dochter.

- **Adjunct en Dependency Principle**: De waarde van MODS op de moeder is de concatenatie van de waarde van MODS op het hoofd en de DT-waarde van een eventuele *modifier* dochter. De waarde van alle andere attributen onder DT is identiek op moeder en hoofd.[5]

Het *head feature principle* veronderstelt een onderscheid tussen HEAD features en andere attributen. In standaard HPSG wordt dit onderscheid geïmplementeerd door de HEAD features te groeperen onder een attribuut HEAD. In de Alpino-grammatica worden de HEAD features expliciet opgesomd in de definitie van het *head feature principle*.

Een *head-complement structure* is een specialisatie van de *headed structure*, waarin naast het hoofd precies één complementdochter optreedt. In dat geval volgt uit het *valence principle* dat de waarde van SC op de moeder gelijk is aan de waarde van SC op het hoofd, minus de complementdochter. Aangezien er geen (geëxtraponeerde) *filler*-dochters of adjuncten in de structuur aanwezig zijn, zal de waarde van SLASH en EXTRA op de moeder de concatenatie zijn van deze waarden op de dochters, terwijl de waarde van MODS identiek zal zijn op moeder en hoofddochter.

De *head-filler*, *head-extra*, en *head-adjunct structures* zijn specialisaties van de *headed structure*, waarin naast het hoofd een dochter optreedt die fungeert als respectievelijk *filler*, geëxtraponeerde dochter, of als adjunct. De waarde van respectievelijk SLASH, EXTRA, of MODS zal in dat geval verschillen op moeder en hoofd, terwijl de waarde van de overige attributen identiek zal zijn.

Verreweg de meeste concrete regels in de grammatica zijn gedefinieerd als instanties van één van bovengenoemde structuren. In de meeste regels dient alleen gespecificeerd te worden wat de categorie en volgorde van de dochters is, en wat het hoofd is. In de voorbeelden hieronder is het hoofd steeds onderstreept:

(1) a. *head-complement-structure*: $v \rightarrow np\ \underline{v}$
 b. *head-adjunct-structure*: $n \rightarrow ap\ \underline{n}$

---

[3]De unificatie van twee AVM's *A* en *B* is de AVM die precies de informatie bevat die in *A* of *B* aanwezig is. Unificatie mislukt wanneer *A* en *B* tegenstrijdige informatie bevatten.

[4]Zie Bouma et al. (2001) voor meer uitleg en motivatie voor deze *head-driven* benadering van extractie.

[5]Merk op dat adjuncten, in tegenstelling tot de analyse in Van Noord & Bouma (1994) en Bouma, Malouf & Sag (2001), niet in het lexicon geïntrodeerd worden. De lijst MOD onder het attribuut DT dient alleen om de dependentierelatie tussen een adjunct en een hoofd weer te geven, en beregelt niet de selectie van adjuncten.

Toevoegen van de informatie die uit de principes kan worden afgeleid aan de regels in (1) (in de implementatie gebeurt dit automatisch tijdens het compileren van de regels (*off-line*)), levert de regels op in (2), waarbij de attributen NFORM en VFORM als voorbeeld van een HEAD-feature fungeren. $\langle H|R\rangle$ staat hier voor een lijst met als eerste element H en als staart R, $L \oplus M$ representeert de concatenatie van twee lijsten L en M.

(2) a.

$$\begin{bmatrix} \textit{verb} & \\ \text{VFORM} & \boxed{1} \\ \text{SC} & \boxed{2} \\ \text{SLASH} & \boxed{3} \\ \text{EXTRA} & \boxed{4} \oplus \boxed{5} \\ \text{DT} & \boxed{6} \end{bmatrix} \rightarrow \boxed{7} \begin{bmatrix} \textit{noun} & \\ \text{EXTRA} & \boxed{4} \end{bmatrix} \begin{bmatrix} \text{VFORM} & \boxed{1} \\ \text{SC} & \langle \boxed{7} | \boxed{2} \rangle \\ \text{SLASH} & \boxed{3} \\ \text{EXTRA} & \boxed{5} \\ \text{DT} & \boxed{6} \end{bmatrix}$$

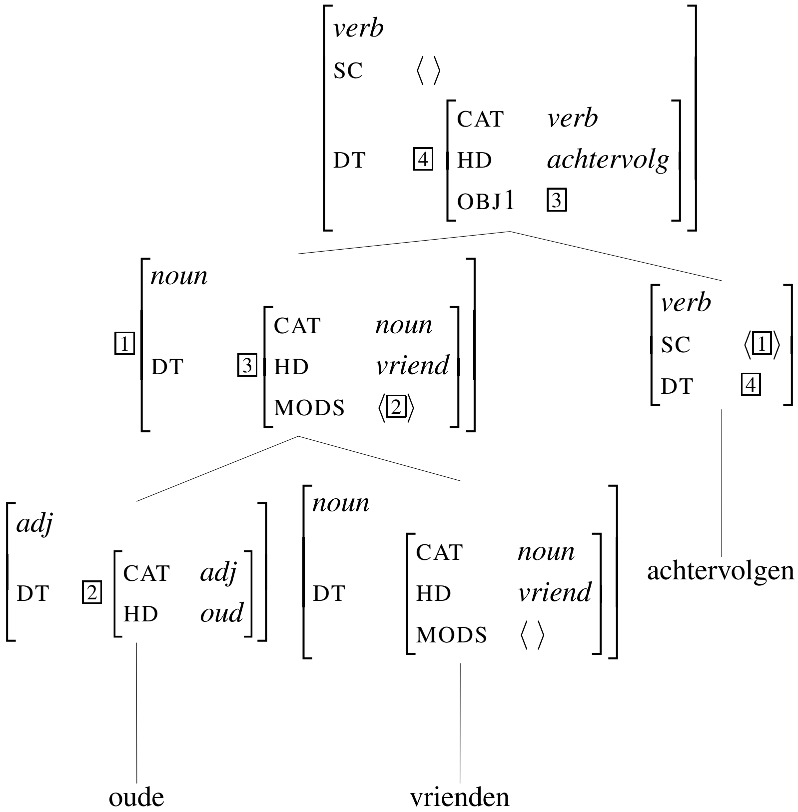
b.

$$\begin{bmatrix} \textit{noun} & \\ \text{NFORM} & \boxed{1} \\ \text{SC} & \boxed{2} \\ \text{SLASH} & \boxed{3} \\ \text{EXTRA} & \boxed{4} \oplus \boxed{5} \\ \text{DT} & \begin{bmatrix} \text{CAT} & \boxed{6} \\ \text{HD} & \boxed{7} \\ \text{MODS} & \langle \boxed{9} | \boxed{8} \rangle \end{bmatrix} \end{bmatrix} \rightarrow \begin{bmatrix} \textit{adj} & \\ \text{EXTRA} & \boxed{4} \\ \text{DT} & \boxed{9} \end{bmatrix} \begin{bmatrix} \textit{noun} & \\ \text{NFORM} & \boxed{1} \\ \text{SC} & \boxed{2} \\ \text{SLASH} & \boxed{3} \\ \text{EXTRA} & \boxed{5} \\ \text{DT} & \begin{bmatrix} \text{CAT} & \boxed{6} \\ \text{HD} & \boxed{7} \\ \text{MODS} & \boxed{8} \end{bmatrix} \end{bmatrix}$$

De regels maken het mogelijk voor de VP in (3a) de structuur in (3b) af te leiden (waarbij attributen die geen rol spelen zijn weggelaten). De constituent *oude vrienden* wordt afgeleid met behulp van regel (2b) en de VP met behulp van regel (2a).

(3) a. (Kim blijft) oude vrienden achtervolgen

b.



De regels in (4) maken het mogelijk een nominale of een verbale constituent te combineren met een geëxtraponeerde bijzin.

(4) a. *head-extra-structure*: $n \rightarrow \underline{n}\ sbar$

b.
$$\begin{bmatrix} \textit{noun} & \\ \text{AGR} & \boxed{1} \\ \text{SC} & \boxed{2} \\ \text{SLASH} & \boxed{3} \\ \text{EXTRA} & \boxed{4} \\ \text{DT} & \boxed{5} \end{bmatrix} \rightarrow \begin{bmatrix} \textit{noun} & \\ \text{AGR} & \boxed{1} \\ \text{SC} & \boxed{2} \\ \text{SLASH} & \boxed{3} \\ \text{EXTRA} & \langle \boxed{6} | \boxed{4} \rangle \\ \text{DT} & \boxed{5} \end{bmatrix} \boxed{6}\begin{bmatrix} \textit{sbar} \end{bmatrix}$$

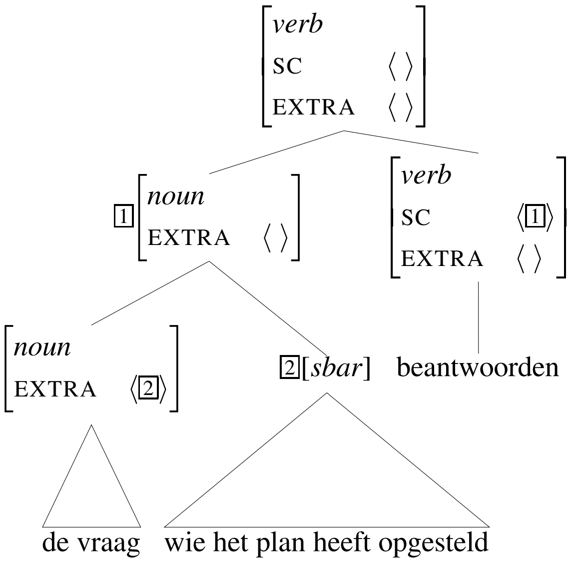
c. *head-extra-structure*: $v \rightarrow \underline{v}\ sbar$

d.
$$\begin{bmatrix} \textit{verb} & \\ \text{AGR} & \boxed{1} \\ \text{SC} & \boxed{2} \\ \text{SLASH} & \boxed{3} \\ \text{EXTRA} & \boxed{4} \\ \text{DT} & \boxed{5} \end{bmatrix} \rightarrow \begin{bmatrix} \textit{verb} & \\ \text{AGR} & \boxed{1} \\ \text{SC} & \boxed{2} \\ \text{SLASH} & \boxed{3} \\ \text{EXTRA} & \langle \boxed{6} | \boxed{4} \rangle \\ \text{DT} & \boxed{5} \end{bmatrix} \boxed{6}\begin{bmatrix} \textit{sbar} \end{bmatrix}$$

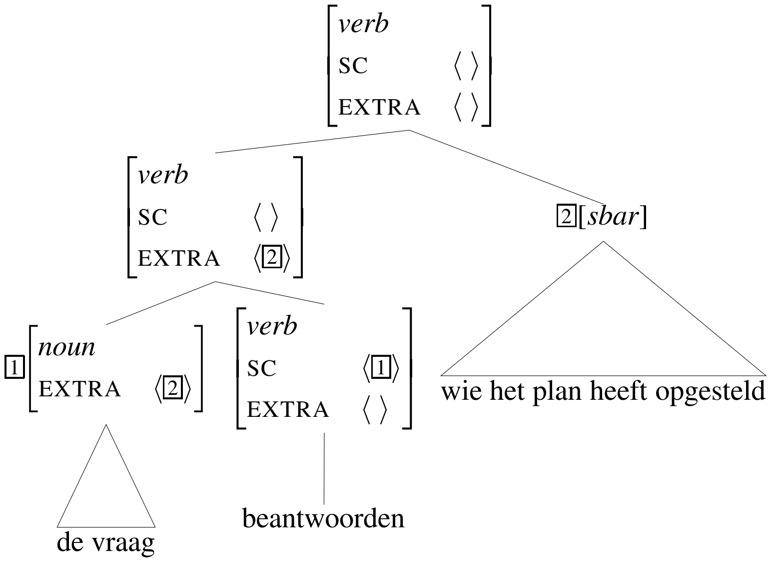
Met behulp van deze regels kunnen de constituenten in (5) worden afgeleid.

(5) a. (Kim moet) de vraag wie het plan heeft opgesteld beantwoorden

b. (Kim moet) de vraag beantwoorden wie het plan heeft opgesteld c.



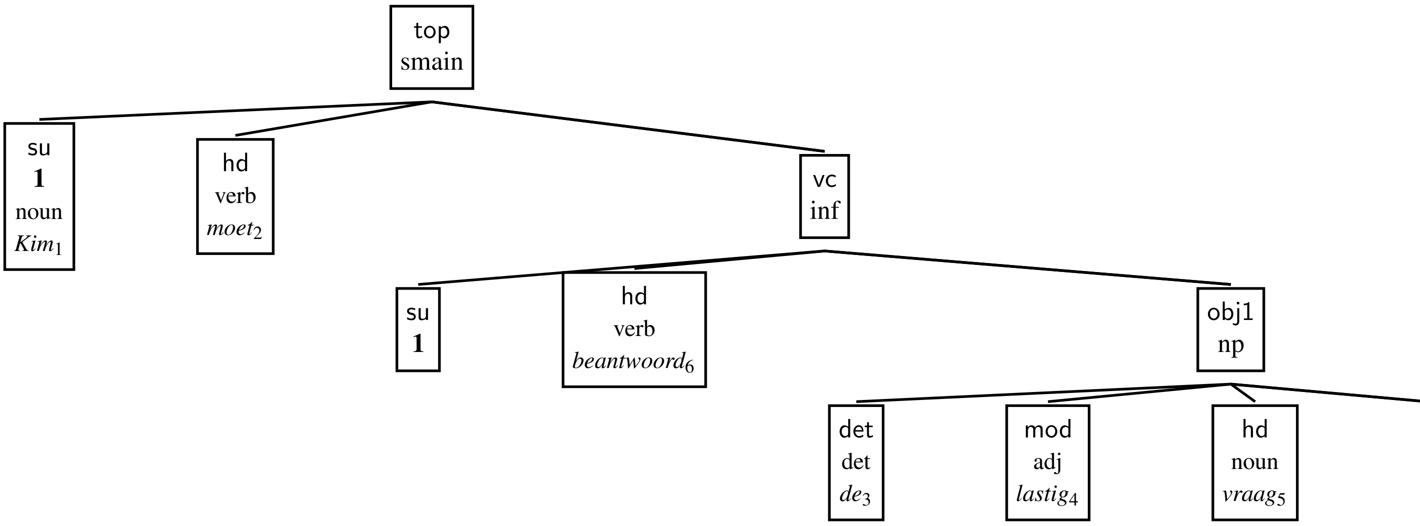
d.



De betekenis van woorden en zinnen wordt in HPSG normaal gesproken weergegeven door semantische representaties aan woorden en regels toe te voegen, samen met principes die de constructie van zulke representaties definiëren. De Alpino-grammatica kent momenteel geen semantische component. Wel worden de grammaticale dependentierelaties gecodeerd, in de vorm van een dependentiestructuur.

Een dependentiestructuur geeft de grammaticale relaties binnen een zin of zinsdeel weer. Een voorbeeld wordt gegeven in figuur 2. De knopen in de boom bestaan uit een grammaticale relatie (*hd* voor *hoofd*, *su* voor *subject*, *mod* voor *modifier*, etc.), eventueel een index (de index **1** geeft aan dat het subject van *moet* identiek is aan het subject van *beantwoord*), en een syntactische categorie. Bladeren bestaan uit een grammaticale relatie, een syntactische categorie, en de stam van een woord met een subscript dat correspondeert met de positie in de zin, of uit een grammaticale relatie en een index. Een belangrijke eigenschap van dependentiestructuur is het feit dat ze abstraheren van woordvolgorde. De discontinue string *de vraag ... of Anne komt* kan dus als een eenheid worden weergegeven, ondanks het feit dat het complement is geëxtraponeerd.

Dependentiestructuren worden opgebouwd door aan regels en *lexical entries* een attribuut DT



Figuur 2: Dependency tree voor de zin Kim moet de lastige vraag beantwoorden of Anne komt.

(voor *dependency tree*) toe te voegen. Werkwoorden hebben een DT-waarde waarin is gedefinieerd met welke grammaticale relaties de complementen van het werkwoord corresponderen. De infinitiefvorm van het werkwoord *beantwoorden* selecteert bijvoorbeeld voor een direct object (OBJ1). Dit betekent dat de DT-waarde van dit werkwoord gedefinieerd kan worden als in figuur 1. Uit de DT-waarde voor een zin kan eenvoudig een representatie als in figuur 2 worden afgeleid. De voornaamste redenen om te kiezen voor dependentiestructuren is dat dit een geschikt formaat lijkt om op een theorieneutrale en effectieve wijze een corpus te annoteren met grammaticale structuren (Skut, Krenn & Uszkoreit 1997). Een belangrijk bijkomstig argument is dat op deze wijze kan worden aangesloten bij de annotatierichtlijnen van het Corpus Gesproken Nederlands (Moortgat, Schuurman & Van der Wouden 2000).

3 De Alpino-grammatica

De Alpino-grammatica bestaat uit een verzameling regels, gebaseerd op de principes die we hierboven hebben geschetst, en uit een lexicon. Samen beschrijven ze een niet-triviaal deel van de grammatica van het Nederlands. In paragraaf 5 proberen we de dekking van de grammatica preciezer te bepalen.

3.1 Lexicon

Grammaticale structuren worden in HPSG voor een belangrijk deel bepaald door de lexicale eigenschappen van de woorden, en met name de syntactische hoofden, die deel uitmaken van zo'n structuur. De accuratesse en reikwijdte van de grammatica wordt daarom voor een groot deel mede bepaald door de omvang van het lexicon en de mate van detail waarmee lemma's worden gedefinieerd.

Het Alpino-lexicon bevat momenteel meer dan 70 verschillende verbale valentiepatronen. Een valentiepatroon definieert voor welke complementen, inclusief het subject bij werkwoorden, een hoofd selecteert. De waarde van SC van een woord wordt bepaald door het valentiepatroon voor dit woord. De meeste patronen kunnen ook gebruikt worden met een scheidbaar prefix.[6] Een overzicht van de distributie van verbale valentiepatronen in het lexicon wordt gegeven in tabel 1. Bij de opbouw van het lexicon hebben we gebruik gemaakt van lexicale informatie die voorhanden is in de lexicale databases van Celex (Baayen, Piepenbrock & Van Rijn 1993), Parole,[7] en het CGN (Groot 2000). De Celex-database is vooral nuttig omdat het de verschillende morfologische vormen van een groot aantal bijvoeglijke en zelfstandige naamwoorden en werkwoorden bevat. Valentiepatronen zijn ontleend aan de lexica van Parole en CGN. Beide lexica specificeren de categorie en grammaticale functie van complementen. Voor het Alpino-lexicon wordt gebruik gemaakt van de vereniging van valentiepatronen uit deze twee bronnen (Bouma 2001). Daarnaast bevat het lexicon valentiepatronen die we zelf hebben toegevoegd. Momenteel zijn dit vooral patronen voor de verschillende hulpwerkwoorden, en voor idiomatische uitdrukkingen zoals *het pleit beslechten, als de dood zijn voor iets, het eens zijn met iets, op de been blijven, in staat zijn,* etc. Voor bijvoeglijke en zelfstandige naamwoorden zijn een veel kleiner aantal valentiepatronen beschikbaar, voornamelijk voor de selectie van verbale complementen. De betreffende lemma's zijn ontleend aan Parole.

Het Alpino-lexicon bevat zo'n 47.000 lemma's. Het streven is de meest frequente woorden met hun syntactische eigenschappen in het woordenboek op te nemen. Voor het ontleden van vrije tekst (bijvoorbeeld journalistiek proza) betekent dit desalniettemin dat er met een zekere regelmaat woorden voorkomen die niet in het woordenboek staan. Naast eigennamen gaat het hierbij vooral om samenstellingen. De component voor lexicale analyse voorspelt voor woorden die niet in het woordenboek staan een mogelijke categorie op basis van heuristieken die bijvoorbeeld in aanmerking nemen of een woord met een hoofdletter begint, of het woord kan worden geanalyseerd als een samenstelling, etc.

3.2 Regels

De grammatica bevat momenteel ongeveer 330 regels. Bijna de helft van de regels zijn *head-modifier* of *head-complement structures*. De andere helft bestaat uit *head-filler-structures* (voor topicalisatie, WH-vragen, en relatieve zinnen), *head-extra-structures* (voor extrapositie van relatieve zinnen, complementzinnen en VP's, PP's, en comparatiefzinnen), regels voor coördinatie, apposities, verbale constituenten ingeleid door een *complementizer*, en enkele regels voor gesproken taal.

De volgende voorbeelden dienen om een globaal overzicht te verkrijgen van de reikwijdte en mate van detail van de regels.

Complementatie. De regels in (6) definiëren welke complementen respectievelijk links en rechts van een verbale projectie kunnen voorkomen. Het label *v-arg(left)* staat voor de disjunctie van NP, PP, en AP. Het label *v-arg(right)* staat voor de disjunctie van $\overline{\text{S}}$, PP, en VP.

(6)
a. *head-complement-structure* : $v \rightarrow$ *v-arg(left)* $\underline{v}$
b. een boek *kopen*, in Sinterklaas *geloven*, aardig *vinden*
c. *head-complement-structure* : $v \rightarrow \underline{v}$ *v-arg(right)*
d. *geloven* dat Sinterklaas bestaat, *geloven* in Sinterklaas, *proberen* om te komen

De regels in (7) definiëren de mogelijke complementen van preposities, waarbij *p-arg* staat voor de

[6] Scheidbare prefixen worden in de grammatica anders behandeld dan complementen, omdat ze geïncorporeerd kunnen zijn in de werkwoordsvorm (*opbelt*), en omdat ze, in tegenstelling tot andere complementen, deel uit kunnen maken van het werkwoordscluster (*heb uit kunnen slapen*).

[7] `http://www.inl.nl/corp/parole.htm`

Valentiepatroon	Aantal	Voorbeeld
[SU:NP][OBJ1:NP]	3438	zij *aanvaardt* het plan
		hij *bakent* het plan *af*
[SU:NP]	2158	zij *aarzelt*
		hij *barst los*
[SU:NP][LD:PP⟨*pform*⟩]	1389	zij *arriveert* in Groningen
		hij *blijft weg* uit Groningen
[SU:NP][PC:PP⟨*pform*⟩]	1271	zij *ageert* tegen het plan
		hij *barst* in tranen *uit*
[SU:NP][OBJ1:NP][LD:PP⟨*pform*⟩]	1013	zij *aait* hem over de bol
		hij *brengt* de kinderen *onder* bij de buren
[SU:NP][OBJ1:NP][PC:PP⟨*pform*⟩]	855	zij *achtervolgt* hem met het plan
		hij *bereidt* haar op het plan *voor*
[SU:NP][OBJ1:SDAT]	418	zij *aanvaardt* dat het plan mislukt
		hij *biecht op* dat het plan mislukt.
[SU:NP][OBJ2:NP][OBJ1:NP]	314	zij *belemmert* hem de doorgang
		hij *biedt* haar het plan *aan*
[SU:SDAT][OBJ1:NP]	274	dat het plan kan mislukken *benauwt* haar
		dat het plan kan mislukken *brengt* onrust *teweeg*
[SU:NP][SE:NP][PC:PP⟨*pform*⟩]	248	zij *baseert* zich op dit plan
		hij *geeft* zich *over* aan de politie

Tabel 1: De meest frequente verbale valentiepatronen in het Alpino-lexicon. Patronen worden gespecificeerd als een lijst complementen (inclusief het subject), en complementen worden gespecificeerd als grammaticale functie:categorie. De functie LD *staat voor locatief of directioneel complement,* SE *voor verplicht reflexief complement.*

disjunctie van NP[NFORM *norm*], PP, AP, $\overline{\text{S}}$, en VP. Regel (7c) is nodig voor gevallen waar naast een prepositie een partikel aanwezig is en (7e) voor zogenaamde *postposities*.

(7) a. *head-complement-structure* : $p \rightarrow \underline{p}$ *p-arg*
b. *in* Groningen, *tot* aan de rand, *op* rood, *zonder* dat het opvalt, *zonder* te twijfelen
c. *head-complement-structure* : $p \rightarrow \underline{p}$ *p-arg part*
d. *naar* Groningen toe
e. *head-complement-structure* : $p \rightarrow$ *p-arg* $\underline{p}$ *part*
f. het dak *op*, het bos *in*

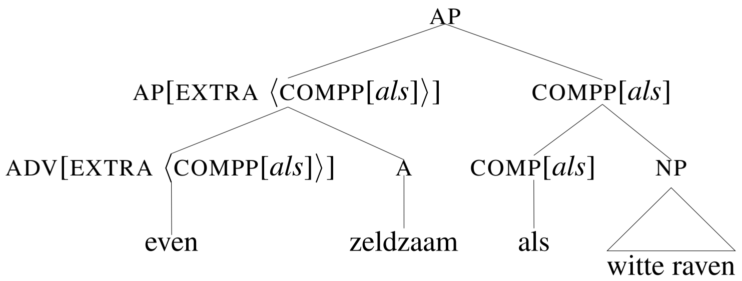
Comparatieven. Bepalingen van vergelijking (ingeleid door *dan* of *als*) zijn optionele complementen die niet noodzakelijkerwijs direct naast het hoofd staan dat deze bepaling selecteert. Adjectieven in de vergrotende trap (8a)-(8c), de adjectieven (die ook adverbiaal gebruikt kunnen worden) *meer* en *minder* (8d)-(8g), en *(n)iets, (n)iemand), (n)ergens anders, niets/niks* (8h)-(8i) kunnen met een *dan*-bepaling optreden. De combinaties *even + adjectief* (8j)-(8l), *(net) zo + adjectief* (8m)-(8o), het bijwoord *evenveel* (8p), of een NP met *hetzelfde* of *dezelfde* (8q)-(8s) kunnen optreden met een *als*-bepaling.

(8) a. Deze prijs ligt *dichter* bij het bod van Bayer *dan* bij dat van Petrofina
b. De internationale fondsen waren aan het slot alle *lager dan* bij opening

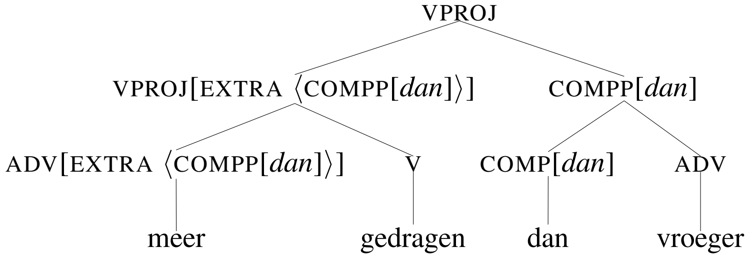
c. Zeker nu hij tien kilo *lichter* is *dan* op de dag dat hij bij de Deventerkernploeg werd ingelijfd
d. De gemeenten verkopen *meer* grond *dan* zij kunnen aankopen
e. Leo is *minder* ijverig dan zijn broertje
f. Hoeden worden *meer* gedragen *dan* vroeger
g. Een programma waarmee hij zich als artiest *minder* kon afficheren *dan* als vakman
h. *Niks anders* doen *dan* almaar ruw materiaal verzamelen
i. Over van Gaal *niets dan* lof
j. Deze koersen zijn *even* zeldzaam *als* witte raven
k. *Even* duister en ondoorgrondelijk *als* het feit dat het kennelijk het verkeerde resumé was
l. *Even* belangrijk *als* een goed in elkaar getimmerd partijprogramma
m. Tenminste driemaal *zo* groot *als* tien jaar geleden
n. Niet meer *zo* goed *als* vroeger
o. Maar wij zijn *net zo goed* machteloos *als* de regering en de Verenigde Naties
p. Uitgeschakeld worden voor het Jaarbeursstedentoernooi zou *evenveel* betekenen *als* niet meer meetellen in het internationale voetbal
q. Jonge mensen uit *dezelfde* leeftijdscategorie *als* de werkende jongeren
r. Dat was *dezelfde als* gisteren
s. Daar zat *hetzelfde* idee achter *als* bij 't aderlaten

In de grammatica wordt aan lexicale elementen die met een bepaling van vergelijking kunnen optreden, onder andere een definitie toegekend waarin EXTRA een comparatief (COMPP) bevat. Omdat extrapositie is toegestaan op het niveau van AP, NP, en VP, voorspellen we onder meer de mogelijkheid van extrapositie binnen een (predikatieve) adjectivische constituent (9a) en binnen een VP (9b).

(9) a.



b.



De bepaling van vergelijking zelf wordt gevormd door het voegwoord *als* of *dan* gevolgd door een NP, bijwoord, bijzin, VP, A (*witter dan wit*) of PP.

Merk op dat de implementatie van deze analyse gebruik maakt van het feit dat alle regels moeten voldoen aan het *extraposition principle*, en dus een waarde toekennen aan het attribuut EXTRA. Om extrapositie van comparatiefzinnen mogelijk te maken hoeven we dus alleen maar aan te nemen dat adjectieven in de vergrotende trap en een aantal specifieke lexicale items via het attribuut EXTRA selecteren voor een bepaling van vergelijking. Een bijkomend voordeel van de analyse in termen van extrapositie is dat ze de gevallen subsumeert waar de bepaling samen met het hoofd een constituent vormt (*lager dan bij de opening was de koers nog nooit, niets dan lof was er voor van Gaal*), en dat ze het verplichte karakter van extrapositie bij bijwoorden als *even* en *zo* verklaart (**even als witte raven zeldzaam,* **zo als vroeger goed*) omdat extrapositie op het niveau van ADVP immers niet is toegestaan.[8]

Modificatie van nomina. In (10) worden enkele regels voor modificatie van nominale constituenten gegeven. Bij modificatie van nomina wordt een onderscheid gemaakt tussen modificatie door een adjectief, PP, of relatieve zin, en apposities. Omdat apposities met de dependentierelatie APP worden gemarkeerd, is voor de betreffende regels een *head-app-structure* gecreëerd. De categorie *app_n* definieert welke nomina als hoofd kunnen optreden in constructies als *een zak aardappelen*. Het gaat hier in het bijzonder om maataanduiders.

(10) a. *head-adjunct-structure : n → n pp*
b. *familie* uit Amsterdam
c. *head-adjunct-structure : n → n rel*
d. *familie* die niemand kent
e. *head-adjunct-structure : np → np post-np-adv*
f. *Beerta* senior/alleen/zelf/ook, *2 februari* aanstaande,
g. *head-adjunct-structure : n → pn n*
h. Chevrolet *programma*
i. *head-app-structure : n → app_n n*
j. een *zak* aardappelen, het *medium* film,
k. *head-app-structure : n → n np*
l. de *familie* Balemans, de *Oostenrijker* Hermann Nitsch, de *hoofdstad* Luxemburg

Partitieve genitieven. Een bijzondere vorm van modificatie zijn partitieve genitieven, die bestaan uit een nominale kern gevolgd door een adjectief met genitief *-s*:

(11) a. 't Is me *wat moois*
b. Dat belooft *niet veel goeds*
c. *Wat voor stoms* heb je nu weer uitgehaald?
d. Het is *niets bijzonders*

De nomina die selecteren voor deze constructie (*iets, wat, niets, niks, niet veel, weinig, genoeg, allerlei, meer* of *wat voor*) vormen een kleine, gesloten klasse. In het lexicon zijn deze nomina van het type *iets_n* (12a), dat alleen voor kan komen in de regel voor partitieve genitieven (12b). De regel verlangt daarnaast een adjectief met *-s*-uitgang als adjunct dochter. Deze vorm van het adjectief wordt geconstrueerd tijdens de compilatie van het lexicon. Ze onderscheiden zich van attributieve en predikatieve adjectieven door de waarde *iets* voor het attribuut AFORM (12c).

[8]Extrapositie vanuit een getopicaliseerde constituent is in het algemeen niet uitgesloten: *De vraag is gerechtvaardigd waarom de regering niets doet*. Een beoordelaar wees erop dat dit bij comparatieven een minder goed resultaat geeft: **Lager was de koers nog nooit dan bij opening*. Omdat de grammatica geen onderscheid maakt tussen extrapositie van bepalingen van vergelijking en andere vormen van extrapositie, worden dergelijke zinnen momenteel door het systeem geaccepteerd. Hiervoor is ook wel wat te zeggen. Ten eerste blijken dergelijke voorbeelden in corpora wel voor te komen: *Liever benadrukt hij die tegenstellingen dan de bedrieglijke harmonie* en *Nog eerder zal de machtige Mekong droogvallen dan dat de co-premier zijn macht uit handen geeft (Volkskrant 1997)*. Daarnaast worden in de grammatica *zo ADJ ... dat* constructies ook als comparatieven behandeld (zoals voorgeschreven door CGN); deze constructie laat extrapositie uit topicalisatie heel gemakkelijk toe: *Zo intens lelijk zijn mijn voeten in de loop van een decennium geworden dat ik de mensenmassa's op het strand er in de zomer niet mee wil lastigvallen (Volkskrant 1997).*

(12) a. $iets$: $\begin{bmatrix} \textit{iets\_n} & \\ \textsc{sc} & \langle\, \rangle \\ \textsc{nform} & \textit{norm} \end{bmatrix}$

b. *head-adjunct-structure: np* → <u>*iets_n*</u> *iets_adj*

c. $lekkers$: $\begin{bmatrix} \textit{adj} & \\ \textsc{aform} & \textit{iets} \end{bmatrix}$

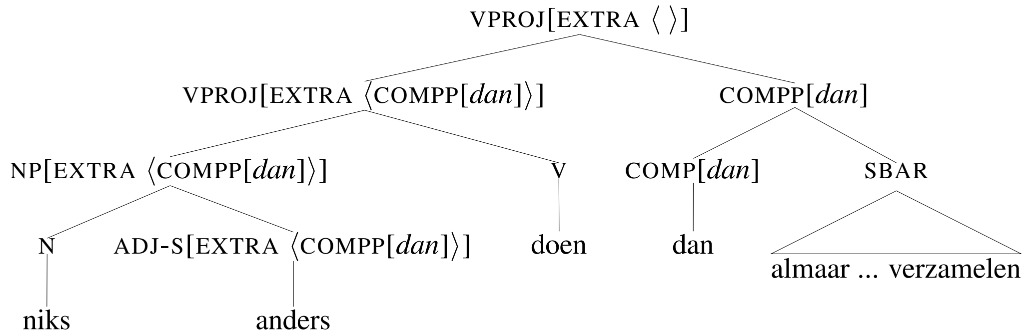
De analyse van de partitieve genitief vereist een aantal tamelijk constructiespecifieke stipulaties, maar maakt toch zo veel mogelijk gebruik van de algemene principes en lexicale categorieën van de grammatica. Doordat de adjectieven in deze constructie zich alleen in het attribuut AFORM van andere adjectieven onderscheiden, kunnen twee genitieve adjectieven bijvoorbeeld gecoördineerd worden met behulp van de algemene regel voor coördinatie van adjectieven:

(13) invallen, waarin ritme en melodie samenvloeiden tot *iets moois maar grilligs*

Een voorbeeld als (14a), tenslotte, demonstreert de interactie van specifieke regels met algemene principes. De NP *niks anders* wordt geanalyseerd als een partitieve genitief. Omdat de regel voor partitieve genitieven een instantie is van een *head-adjunct-structure*, wordt de waarde van EXTRA op de juiste manier doorgegeven, en wordt zonder extra stipulaties voorspeld dat de bepaling van vergelijking die *anders* introduceert vanuit deze constructie geëxtraponeerd kan worden.

(14) a. *Niks anders* doen *dan* almaar ruw materiaal verzamelen

b.



Overige Regels. Een beschrijving van de manier waarop enkele lastige aspecten van de Nederlandse syntaxis worden verantwoord, valt buiten het bereik van dit artikel. We volstaan daarom met enkele verwijzingen naar eerder werk. In navolging van Koster (1975) relateren we het finiete werkwoord in hoofdzinnen aan de VP-finale positie waar in bijzinnen het finiete werkwoord te vinden is. De non-transformationele implementatie van dit idee wordt besproken in Van Noord, Bouma, Koeling & Nederhof (1999). Een niet onbelangrijk aspect van de grammatica van het Nederlands is de analyse van verbale eindclusters. Hier maken we gebruik van *argument inheritance* (Hinrichs & Nakazawa 1994). Een verschil met het voorstel in Bouma & Van Noord (1997) is dat in de Alpino-grammatica geen gebruik wordt gemaakt van zogenaamde *linear precedence constraints*, maar dat de woordvolgorde binnen het cluster wordt beschreven door herschrijfregels. De computationele nadelen van *linear precedence constraints* wegen in dit geval zwaarder dan het feit dat hierdoor wellicht enkele generalisaties worden gemist. Constituentvragen en relatieve zinnen maken gebruik van de analyse van extractie zoals voorgesteld in Bouma, Malouf & Sag (2001), met uitzondering van extractie van adjuncten, waarvoor, wederom uit computationele overwegingen, een syntactische in plaats van lexicale analyse is geïmplementeerd.

4 Corpusannotatie en Evaluatie

De Alpino-grammatica wordt als hulpmiddel gebruikt bij het construeren van een collectie syntactisch geannoteerde Nederlandse zinnen, de *Alpino Treebank*. De annotatie bestaat, in navolging van het Corpus Gesproken Nederlands (Moortgat et al. 2000), uit dependentiestructuren. In paragraaf 4.1 wordt kort besproken hoe hierbij te werk wordt gegaan.

Het geannoteerde corpus wordt gebruikt om de kwaliteit van het systeem nauwkeurig te kunnen volgen. Hiertoe wordt het systeem toegepast op de zinnen uit het corpus. De door het systeem als beste beschouwde analyse wordt vervolgens systematisch vergeleken met de geannoteerde versie; deze vergelijking leidt vervolgens tot een kwantitatieve beoordeling van het systeem. Door deze methode kan bijvoorbeeld worden vastgesteld dat toevoegingen aan de grammatica of veranderingen in de grammatica geen onbedoelde problemen veroorzaken. In paragraaf 4.2 laten we zien hoe de *treebank* voor evaluatiedoeleinden wordt ingezet.

Evaluatie aan de hand van de beste analyse die door het systeem wordt geproduceerd veronderstelt een methode om de kwaliteit van analyses te beoordelen. De Alpino Treebank wordt onder andere gebruikt om een statistisch model af te leiden dat in staat is de beste analyse te kiezen uit een verzameling mogelijke analyses. In paragraaf 4.3 geven we een korte schets van de techniek die we hiervoor gebruiken.

4.1 De constructie van de Alpino Treebank

De constructie van de *treebank* behelst het annoteren van een gegeven zin met CGN dependentiestructuren. De annotatie verloopt als volgt. De zin wordt door het systeem automatisch geanalyseerd. Hierbij heeft de annotator de mogelijkheid om in te grijpen in het parseerproces door de lexicale analyse van woorden handmatig te bepalen. Ook kan door middel van het plaatsen van haakjes in de zin de parser in de juiste richting worden gedwongen.

De parser levert in het algemene geval een groot aantal analyses op. Een selectieprogramma, gebaseerd op de SRI TreeBanker (Carter 1997), stelt de annotator in staat snel de juiste analyse te kiezen. Het is hierbij niet nodig de verschillende analyses allemaal één voor één te bekijken. Het komt natuurlijk ook regelmatig voor dat geen enkele analyse de juiste is. In dat geval kiest de annotator een analyse die zo min mogelijk afwijkt van de correcte analyse, en past deze analyse vervolgens handmatig aan met *Thistle* (Calder 2000). Thistle is een programma voor het bewerken en visualiseren van taalkundige structuren. Tenslotte wordt de correct geachte dependentiestructuur als XML-code opgeslagen, en later nog gecontroleerd en mogelijk gecorrigeerd door een tweede annotator.

Momenteel zijn alle 7150 zinnen van het dagbladendeel van het Eindhoven corpus geannoteerd (Uit den Boogaart, 1975). Daarnaast zijn nog enkele kleinere verzamelingen zinnen geannoteerd voor het testen van de grammatica (o.a. de voorbeeldzinnen uit de CGN handleidingen, en voorbeeldzinnen die bij het ontwikkelen van de grammatica zijn gehanteerd). [9]

4.2 Evaluatie

Het syntactisch geannoteerde corpus wordt gebruikt voor evaluatie van de grammatica. Om vast te stellen in hoeverre een door het systeem geproduceerde dependentiestructuur correct is, bepalen we de afhankelijkheidsrelaties die in zo'n structuur aanwezig is. De dependentiestructuur voor de zin in (15a), gegeven in figuur 2, levert de verzameling relaties in (15b) op.

(15) a. Kim moet de lastige vraag beantwoorden of Anne komt

[9] De huidige versie van de *treebank* kan worden geraadpleegd op `http://www.let.rug.nl/~vannoord/trees/`

b.	⟨ moet	su	Kim ⟩	⟨ moet	vc	beantwoord ⟩
	⟨ beantwoord	su	Kim ⟩	⟨ beantwoord	obj1	vraag ⟩
	⟨ vraag	det	de ⟩	⟨ vraag	mod	lastig ⟩
	⟨ vraag	vc	of ⟩	⟨ of	body	kom ⟩
	⟨ kom	su	Anne ⟩			

Relaties bestaan steeds uit drie delen: het woord dat correspondeert met een syntactisch hoofd, de naam van de relatie, en het hoofd van de constituent die in de genoemde relatie tot het eerste woord staat.

Voor evaluatie wordt het aantal relaties van de beste door het systeem opgeleverde analyse (D_s) geteld, en van de analyse in de *treebank* (D_t). D_f is het aantal foute en ontbrekende relaties in de door het systeem opgeleverde analyse. De *accuratesse* wordt vervolgens gedefinieerd als

$$accuratesse = 1 - \frac{D_f}{max(D_t, D_s)}$$

Deze formule levert een waarde op tussen 0 (helemaal fout) en 1 (helemaal goed), die grofweg kan worden geïnterpreteerd als het percentage juiste afhankelijkheden.

4.3 Desambiguatie

Evaluatie veronderstelt dat een keuze kan worden gemaakt uit de verschillende analyses van een zin die volgens de grammatica mogelijk zijn.

We maken gebruik van een *log-linear* (*maximum entropy*) statistisch model om een keuze tussen deze analyses te kunnen maken (Johnson, Geman, Canon, Chi & Riezler 1999). Het model telt eigenschappen van een analyse, zogenaamde *features*,[10] die relevant lijken voor desambiguatie. Zo is elke regel in de grammatica een feature, en zijn er features voor de verschillende heuristieken voor het toekennen van taalkundige categorieën aan onbekende woorden. Ook zijn er features die aangeven welke afhankelijkheidsrelaties optreden in een gegeven *dependency structure*. Welke features van belang zijn wordt voorlopig vooral handmatig bepaald (maar zie Mullen (2002)).

Voor een gegeven analyse kunnen we vervolgens bepalen hoe vaak elk mogelijk feature optreedt. In het statistisch model wordt aan elk feature i een gewicht λ_i toegekend. Een positief gewicht suggereert dat een analyse dat dit feature bevat geprefereerd wordt. Een negatief gewicht betekent juist dat het model analyses met zulke features liever niet ziet.[11]

Om een model te construeren moet voor elk feature het corresponderende gewicht worden bepaald. Voor het toekennen van de gewichten gebruiken we een nieuwe implementatie van Maximum

[10]De *features* die gebruikt worden voor desambiguatie zijn willekeurige eigenschappen van een syntactische analyse, en vallen dus niet samen met de *features* of attributen die in de grammatica gebruikt worden.

[11]In zo'n model wordt de kans dat een gegeven zin x de analyse y heeft als volgt gedefinieerd, waarbij f_i staat voor het aantal voorkomens van het feature i:

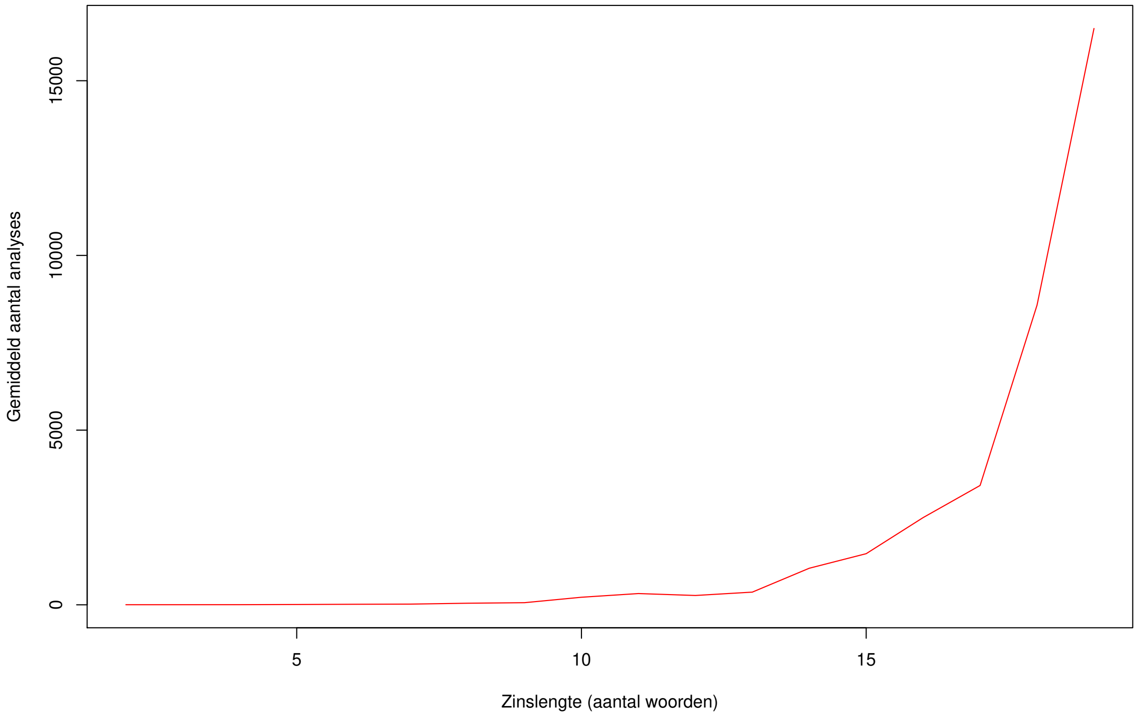
$$p(y|x) = \frac{1}{Z(x)} \exp\left(\sum_i \lambda_i f_i(x,y)\right)$$

De partitiefunctie $Z(x)$ is voor elke parse van x gelijk, en daarom hoeft dit deel van de formule niet te worden berekend: we zijn niet geïnteresseerd in de uiteindelijke waarschijnlijkheid, maar louter in de analyse die de grootste waarschijnlijkheid heeft. Om te bepalen welke analyse de meest waarschijnlijke is hoeven we daarom alleen voor elke analyse de volgende kwantiteit te berekenen:

$$\sum_i \lambda_i f_i(x,y)$$

Entropy van Rob Malouf (Malouf 2002). De input voor deze berekening bestaat uit een aantal analyses waarbij voor elke analyse de aanwezige features zijn gespecificeerd, en voor elke analyse bovendien wordt aangegeven hoe goed de analyse is. Om deze input te produceren wordt de parser (zonder desambiguatiemodel) toegepast op de zinnen van de *treebank*. De bepaling van de kwaliteit van elke analyse gebeurt door de accuratesse te berekenen, zoals uitgelegd in de vorige paragraaf.

Om voor een gegeven zin de juiste analyse te selecteren moet voor elke analyse de door het model toegekende score worden berekend. Het aantal analyses neemt echter exponentieel toe wanneer de lengte van de zin toeneemt. Figuur 3, waarin het gemiddeld aantal analyses is uitgezet voor zinslengtes tot 20 (volgens de huidige versie van de grammatica), laat zien dat dit fenomeen ook een praktisch probleem oplevert. De parser berekent daarom niet expliciet alle analyses maar construeert voor een gegeven zin een zogenaamd *parse forest*: een compacte datastructuur waarin elke mogelijke analyse makkelijk terug te vinden is. Een zoekprocedure zoekt vervolgens in dit *parse forest* naar de *beste* analyse. De procedure evalueert hiertoe ook gedeeltelijke analyses met het desambiguatiemodel. De procedure vindt heel snel een zeer goede, maar niet gegarandeerd de beste analyse.



Figuur 3: Gemiddeld aantal parses per zinslengte

5 Voorlopige resultaten

Om een indruk te geven van de huidige versie van de grammatica bespreken we hier het resultaat van twee experimenten.

In het eerste experiment wordt de accuratesse van de beste analyse volgens Alpino voor de eerste honderd zinnen van het dagbladendeel van het Eindhoven corpus gemeten. Deze honderd zinnen bevatten gemiddeld zo'n 20 woorden, waarbij achttien zinnen meer dan dertig woorden bevatten. Het desambiguatiemodel dat bij dit experiment wordt gebruikt is getraind op zo'n 3000 andere zinnen uit het Eindhoven corpus. In de linker kolom van tabel 2 is te zien dat slechts in negentien gevallen de door het systeem geprefereerde analyse helemaal correct was. Grote delen van de analyse waren meestal wel in orde: het systeem behaalde een gemiddelde accuratesse per zin van bijna tachtig procent; de totale accuratesse (waarbij de accuratesse wordt berekend over de som van de relaties van alle zinnen samen) was ongeveer 78%. In de gemiddelde accuratesse per zin wegen korte en lange zinnen even zwaar, in de totale accuratesse wegen lange zinnen (die moeilijker zijn) zwaarder dan korte zinnen.

100% accuratesse	19.0	77.7
gemiddelde accuratesse per zin	79.8	92.5
totale accuratesse	78.0	94.0

Tabel 2: Accuratesse van de beste analyse volgens Alpino op de eerste honderd zinnen van het Eindhoven corpus (links) en accuratesse van de best mogelijke analyse op 2430 korte zinnen (rechts).

In het eerste experiment meten we de kwaliteiten van zowel de grammatica als de desambiguatiecomponent. Een interessante vraag is in hoeveel van de gevallen de correcte analyse door de grammatica gemaakt wordt, maar niet als hoogste wordt gekenmerkt door de desambiguatiecomponent. Helaas is dit moeilijk vast te stellen omdat voor de langere zinnen miljoenen analyses mogelijk zijn. Om een indruk te geven van de kwaliteit van de grammatica zonder hierbij de desambiguatie te betrekken hebben we daarom een tweede experiment uitgevoerd. Voor alle geannoteerde zinnen van het Eindhoven corpus met ten hoogste vijftien woorden bleek het wel mogelijk alle analyses uit te rekenen. De resultaten zijn te vinden in de rechter kolom van tabel 2.

Uit het tweede experiment blijkt dat, zoals verwacht, desambiguatie een belangrijke invloed op de kwaliteit van het systeem heeft, maar ook is duidelijk dat de grammatica nog voor verbetering vatbaar is. De belangrijkste taalkundige fenomenen uit het corpus die door de grammatica nog niet goed worden behandeld worden hier kort genoemd:

- Ongrammaticale zinnen en spelfouten:

  (16) a. Het EEGtoporgaan heeft gisteravond besloten WestDuitsland zijn beperkende maatregelen tegen de import van landbouwprodukten moet opheffen.
  b. Ruim dertig percent van de tientallen *mijoenen* Japanse tv-kijkers slaat nooit een aflevering van deze wekelijkse show over.

- Onbekende woorden waarvoor heuristieken niet goed werken:

  (17) Langzaamaan werden we bekend.

  Het woord *langzaamaan* werd ten onrechte als nomen geanalyseerd.

- Complexe samenstellingen:

  (18) a. de 8 procent staatsleningen
  b. de 4 x 200 meter ploeg

- Interjecties en andere ingevoegde modificaties (cursief):

(19) a. De Nachtwacht van Rembrandt kun je, *plus hondje*, in levende lijve tegenkomen in Berg en Terblijt.
b. Bij de heren vielen - *maar dat was minder verrassend* - de estafetteformaties volledig door de mand
c. Ook over de wijk waar ik zelf woon *(Buitenveldert in Amsterdam)* worden door lieden, die dit stadsdeel kennelijk slechts oppervlakkig kennen, de meest krasse veroordelingen uitgesproken.

- Onbekende subcategorisatiepatronen:

(20) Dit jaar ziet men zich al voor problemen gesteld.

- PP-complementen van nomina, extrapositie en topicalisatie van PP-complementen van nomina (in het Alpino lexicon ontbreekt tot nu toe de hiervoor benodigde informatie):

(21) a. Bij de minister werd veel begrip gevonden voor de bij de vakbeweging levende wensen.
b. Op bijna alle brieven hebben we geen reacties ontvangen.
c. Van 't woonhuis bleef een groot gedeelte gespaard.
d. Van dat alles bleef niets heel.

- Elliptische constructies; bepaalde vormen van coördinatie; extrapositie van delen van een coördinatie:

(22) a. Tussen Amsterdam en Schiphol zal de lijn ruim twaalf miljoen reizigers per jaar vervoeren, tussen Schiphol en Leiden ruim elf miljoen
b. De vertegenwoordigers van het gas- en electriciteitsbedrijf zouden vandaag en die van de mijnwerkers overmorgen hun stakingsplannen bekend maken [...]
c. Er worden bloembakken gewenst, goede gordijnen, tafelversiering, wandversiering en sfeervolle verlichting.

6 Conclusies

De Alpino-grammatica heeft een veel groter bereik dan eerdere computationele grammatica's voor het Nederlands.[12] Dit is met name te danken aan het feit dat vanaf de start aandacht is besteed aan de prestaties van het systeem op data ontleend aan corpora. Dit heeft ertoe geleid dat er aandacht is besteed aan allerlei vormen van robuustheid, dat er een omvangrijk lexicon is geconstrueerd op basis van algemeen beschikbare elektronische lexica, en dat de grammatica veel gedetailleerder is dan grammatica's die voornamelijk op de taalkundig meest uitdagende constructies zijn gebaseerd.

Veel computationele systemen die geschikt zijn voor het verwerken van grote hoeveelheden tekst beperken zich tot een tamelijk oppervlakkige taalkundige analyse. In de Alpino-grammatica is bewust gekozen voor een formalisme dat alle syntactisch relevante fenomenen kan beschrijven. Zo

[12] Het is moeilijk deze bewering te onderbouwen omdat (voor zover ons bekend is) andere systemen nooit op een vergelijkbare rigoureuze manier zijn geëvalueerd. Bij een recente poging om ontleedsystemen voor het Nederlands te vergelijken (de *"Battle of the Parsers"* tijdens de LOT-winterschool in januari 2001) werd Alpino als winnaar uitgeroepen.

kunnen bijvoorbeeld constituentvragen, relatieve zinnen, extrapositie, subject- en object-controle, en kruisende afhankelijkheden in werkwoordsclusters op een taalkundig verantwoorde manier behandeld worden. Een juiste beschrijving van de afhankelijkheden die in deze constructies optreden is essentieel wanneer het resultaat van de syntactische analyse moet worden weergegeven als een dependentiestructuur. Een bijkomend voordeel van het gebruik van een formalisme dat gebruik maakt van *attribute-value structures* is het feit dat de constructie van zulke dependentiestructuren eenvoudig in de grammatica zelf kan worden geïncorporeerd.

De constructie van een *treebank* met syntactisch geannoteerd materiaal blijkt inmiddels één van de meest waardevolle nevenproducten van het werk aan de grammatica. We zijn ervan overtuigd dat een dergelijke *treebank*, mits van voldoende omvang, van onschatbare waarde is voor het ontwikkelen van computationele grammatica's en voor het berekenen van statistische desambiguatiemodellen. Bovendien zal een dergelijk corpus een welkom hulpmiddel kunnen zijn voor taalkundig onderzoek, omdat ze het mogelijk maakt de syntactische structuren van het Nederlands in geschreven tekst systematisch te onderzoeken.

Referenties

Baayen, R. H., R. Piepenbrock & H. van Rijn (1993). *The CELEX Lexical Database (CD-ROM)*. Philadelphia, PA: Linguistic Data Consortium, University of Pennsylvania.

Uit den Boogaart, P. C. (1975). *Woordfrequenties in geschreven en gesproken Nederlands*. Utrecht: Oosthoek, Scheltema & Holkema.

Bouma, G. (2000). Argument realization and Dutch R-pronouns: Solving Bech's problem without movement or deletion. In: R. Cann, C. Grover & P. Miller (red.), *Grammatical Interfaces in HPSG*, Stanford, CA: CSLI Publications, 51–76.

Bouma, G. (2001). Extracting dependency frames from existing lexical resources. In: *Proceedings of the NAACL Workshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*. Somerset, NJ: Association for Computational Linguistics, 65–70.

Bouma, G., R. Malouf & I. Sag (2001). Satisfying constraints on adjunction and extraction. *Natural Language and Linguistic Theory* 19, 1–65.

Bouma, G. & G. van Noord (1998). Word order constraints on verb clusters in German and Dutch. In: E. Hinrichs, T. Nakazawa & A. Kathol, (red.), *Complex Predicates in Nonderivational Syntax*, New York: Academic Press, 43–72.

Calder, J. (2000). Thistle and interarbora. In: *Proceedings of the 18th International Conference on Computational Linguistics (COLING)*, Saarbrücken, 992–996.

Carter, D. (1997). The TreeBanker: A tool for supervised training of parsed corpora. In: *Proceedings of the ACL Workshop on Computational Environments For Grammar Development And Linguistic Engineering*. Somerset, NJ: Association for Computational Linguistics, 9–15.

Van Eynde, F. (1996). A monostratal treatment of it extraposition without lexical rules. In: W. Daelemans, G. Durieux & S. Gillis (red.), *CLIN 1995, Papers from the sixth CLIN Meeting 1995*. Antwerpen: Universitaire Instelling Antwerpen, 231–248.

Van Eynde, F. (**1999**). Major and minor pronouns in Dutch. In: G. Bouma, E. W. Hinrichs, G.-J. M. Kruijff & R. T. Oehrle (red.), *Constraints and Resources in Natural Language Syntax and Semantics*, Stanford, CA: CSLI Publications, 137–152.

Groot, M. (**2000**). Lexiconopbouw: microstructuur. Intern rapport van het project Corpus Gesproken Nederlands.

Hinrichs, E. & T. Nakazawa (**1994**). Linearizing AUXs in German verbal complexes. In: J. Nerbonne, K. Netter & C. Pollard (red.), *German in Head-driven Phrase Structure Grammar*, Stanford, CA: CSLI Publications, 11–38.

Johnson, M., S. Geman, S. Canon, Z. Chi & S. Riezler (**1999**). Estimators for stochastic "unification-based" grammars. In: *Proceedings of the 37th Annual Meeting of the ACL*, Somerset, NJ: Association for Computational Linguistics, 535–541.

Koster, J. (**1975**). Dutch as an SOV language. *Linguistic Analysis* 1, 111–136.

Malouf, R. (**2002**). A comparison of algorithms for maximum entropy parameter estimation. In: *Proceedings of the Sixth Conference on Natural Language Learning (CoNLL-2002)*. Taiwan.

Moortgat, M., I. Schuurman & T. van der Wouden (**2000**). CGN syntactische annotatie. Intern rapport van het project *Corpus Gesproken Nederlands*.

Mullen, T. (**2002**). *An Investigation into Compositional Features and Feature Merging for Maximum Entropy-Based Parse Selection*. proefschrift, Rijksuniversiteit Groningen.

Van Noord, G. (**1997**). An efficient implementation of the head corner parser. *Computational Linguistics* 23, 425–456.

Van Noord, G. & G. Bouma (**1994**). Adjuncts and the processing of lexical rules. In: *Proceedings of the 15th International Conference on Computational Linguistics (COLING)*, Kyoto, 250–256.

Van Noord, G., G. Bouma, R. Koeling & M.-J. Nederhof (**1999**). Robust grammatical analysis for spoken dialogue systems. *Journal of Natural Language Engineering* 5, 45–93.

Oostdijk, N. (**2000**). Het Corpus Gesproken Nederlands. *Nederlandse Taalkunde* 5, 280–284.

Pollard, C. & I. Sag (**1994**). *Head-driven Phrase Structure Grammar*. Stanford, CA: CSLI Publications.

Sag, I. (**1997**). English relative clause constructions. *Journal of Linguistics* 33, 431–484.

Sag, I. A. & T. Wasow (**1999**). *Syntactic Theory: A Formal Introduction*. Stanford, CA: CSLI Publications.

Skut, W., B. Krenn & H. Uszkoreit (**1997**). An annotation scheme for free word order languages. In: *Proceedings of the Fifth Conference on Applied Natural Language Processing*. Somerset, NJ: Association for Computational Linguistics, 88–95.